REMARKS

This amendment responds to the Final Office Action mailed June 24, 2005. In the office action the Examiner:

- objected to the specification, and in particular informalities in paragraph 10;
- rejected claims 1-14 as being indefinite under 35 U.S.C. 112, first paragraph; and
- rejected claims 1 and 11 under 35 U.S.C. 102(b) as anticipated by Direct Rambus Technology Overview, 1997, pp. 10-12;
- rejected claims 1-14 under 35 U.S.C. 102(b) as being anticipated by Farmwald et al. (US 5,243,703).

After entry of this amendment, the pending claims are: claims 1-14 and 19.

Overview of Changes to the Specification

Informalities in the specification identified by the Examiner in the Final Office Action have been corrected in the present reply. No new matter has been added by these amendments.

Overview of Changes to the Claims

Independent claims 1 and 11 have been clarified to indicate that "during the first time period each memory device in the first plurality of memory devices has a second transfer rate that is less than a first transfer rate for transferring the first data block during the first time period." Support for this amendment is found in the specification on p. 11, paragraph 50. This amendment, therefore, does not constitute new matter.

In addition, the substance of claim 2 has been incorporated into claim 1, and the substance of claim 12 has been incorporated into claim 11. As a result, claims 2 and 12 have been cancelled.

Claim 4 as rewritten and new claim 19 state that, during a first time period, a first memory device transfers a respective second data block during a first portion of a first clock cycle of the first time period, and a second memory device transfers its respective second data block during a second portion of the first clock cycle of the first time period. Support for this amendment is found in the specification in Figure 16 and on page 21, paragraphs 90 and 91. This amendment, therefore, does not constitute new matter.

Claims 1 and 11 have been amended to replace "control information" with a "command." Support for this amendment is found in the specification on p. 12-13, paragraphs 54, 55 and 59. This amendment, therefore, does not constitute new matter.

A number of the other claims have been revised to improve clarity.

Detailed Response 35 U.S.C. 112

After entry of this reply, the limitation in the pending claims 1 and 11 rejected by the Examiner in the present Office Action has been clarified, as described in the preceding paragraph. Support in the specification for these amendments has also been provided. Removal of this ground for rejection is requested.

Detailed Response 35 U.S.C. 102(b)

In the present Office Action the Examiner rejects pending claims 1 and 11 as anticipated by Direct Rambus Technology Overview. The Applicants disagree and traverse.

After entry of this reply, the pending independent claims contain the limitations of a memory controller and a group of memory devices, as a multiplexed group, communicating a first data block during a first time period to the memory controller in accordance with a command from the controller. Each of the memory devices in the group of memory devices has a transfer rate that is less than the transfer rate for transferring a first data block size during a first time period. The Direct Rambus Technology Overview document does not disclose these limitations (i.e., neither the multiplexed group mode of operation for transferring a first data block to a memory controller, nor the transfer rate limitations) and, therefore, does not anticipate these pending independent claims. The Applicants note that in the present Office Action the Examiner has not provided specific support for the assertion that the Direct Rambus Technology Overview document discloses each of the limitations of these pending independent claims. Removal of this ground for rejection is requested.

(The Applicants note that pp. 17-20 of the Direct Rambus Technology Overview document submitted in the reply to the previous Office Action are blank. To the best of the Applicant's knowledge, the Direct Rambus Technology Overview document only contains 16 non-blank pages. A clean version of the same document containing pp. 1-16 is included with this reply.)

With regard to claim 11, it is noted that claim 11 includes a repeater and an auxiliary channel, as well as a group of memory devices that operate as a multiplexed group for

communicating a first data block to a memory controller. This combination is not disclosed in The Direct Rambus Technology Overview document.

In the present Office Action the Examiner also rejects pending claims 1-14 as anticipated by Farmwald (US 5,243,703). The Applicants disagree and traverse.

Since Farmwald discloses neither a multiplexed mode of operation for transferring a first data block from multiple memory devices to a memory controller in response to a command, nor the transfer rate limitations of claims 1 and 11, Farmwald does not anticipate these independent claims.

Since dependent claims include all the limitations of their parent independent claims, Farmwald also does not anticipate dependent claims 2-10, 12-14 and 19. Removal of this ground for rejection is requested.

It is further noted that the time multiplexing aspects of claims 4 and 19, and the repeater and auxiliary aspects of claims 13 and 14 are also not taught by Farmwald and The Direct Rambus Technology Overview.

CONCLUSION

In light of the above amendments and remarks, the Applicant respectfully requests that the Examiner reconsider this application with a view towards allowance. The Examiner is invited to call the undersigned attorney if a telephone call could help resolve any remaining items.

Respectfully submitted,

Date: October 11, 2005

31,066 (Reg. No.)

Gary S. Williams
Morgan, Lewis & Bockius LLP
2 Palo Alto Square
3000 El Camino Real, Suite 700
Palo Alto, California 94306
(650) 843-4000

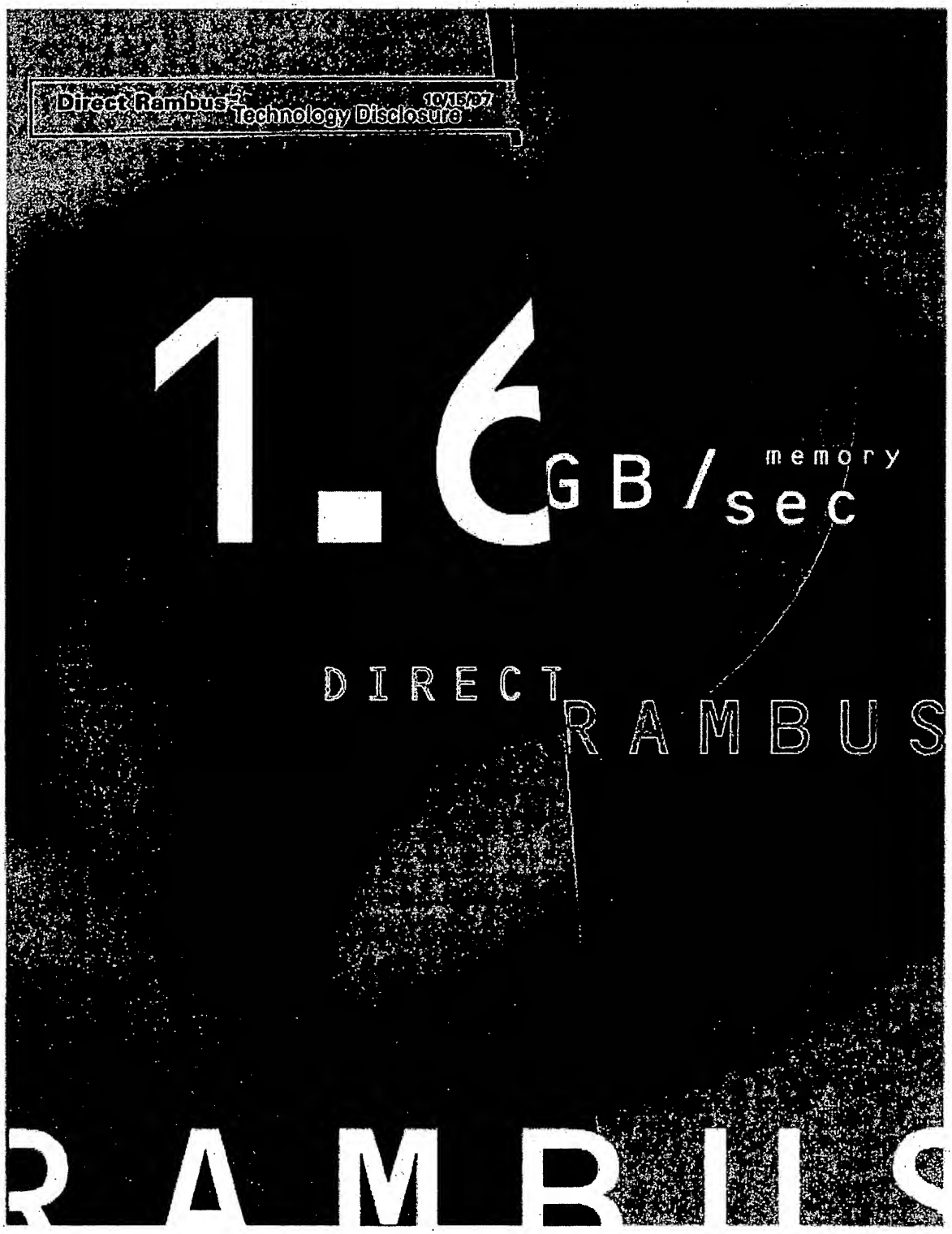

Table of Contents

DL-0040-00

Rambus Inc.
2465 Latham Street
Mountain View, California USA
94040

Telephone: 650-903-3800
Fax: 650-965-1528

www.rambus.com

Preface

The Direct Rambus™ memory technology is an extension of Rambus's high-throughput memory interface technology that is shipping today in PC, workstation and consumer multimedia products. Intended for use in next generation system memory systems, the Direct Rambus technology is ideal for a broad range of systems from consumer digital video products to desktop computers to supercomputers.

The Direct Rambus program goals were set to meet several challenges faced by memory system designers:

1. To help close the processor-memory performance gap, DRAMs need to provide ten times improvement in bandwidth per pin.
2. To be affordable in mainstream markets, the DRAM costs in die size and package must be kept comparable to commodity DRAMs.
3. Apply to a broad range of market segments including consumer, computer, and communications.
4. To provide a stable interface to OEMs, the interface must be able to span multiple DRAM and process generations.

As part of meeting these goals, the following extensions were made to the existing Rambus interface:

- Wider interface: two-byte wide data path
- Higher clock frequency: 800MHz transfer rate
- More efficient protocol: 95% efficiency

As a result, an example Direct Rambus PC main memory system is shown below. It can achieve three times the effective bandwidth over an SDRAM-100 memory system at comparable cost and lower power.

This document introduces the Direct Rambus technology and the devices that implement it. These topics are presented in a series of sections. Each section describes related aspects of Direct Rambus technology and its applications.

The *Introduction* describes the basic technical problem that Rambus Inc. set out to solve and summarizes the resulting solution. The *Advantages of Rambus Technology* and *Applying the Advantages of Rambus Technology* provide summary features and benefits of the technology for typical applications. *The Physical Layer* and *The Logical Layer* define, respectively, the electrical interface and the protocol used between devices on a Direct Rambus Channel. *Inside a Direct Rambus DRAM* describes the internals of a DRAM constructed using Direct Rambus technology. The *Rambus Memory Controller* is digital logic within a Rambus controller used to drive the Rambus protocol. The Rambus *System Packaging* uses industry form factor memory modules and connectors.

Technical specifications, user guides, data sheets and application notes are also available from Rambus Inc. These provide further information about Rambus technology, how to implement systems that use it and individual Rambus-compatible devices.

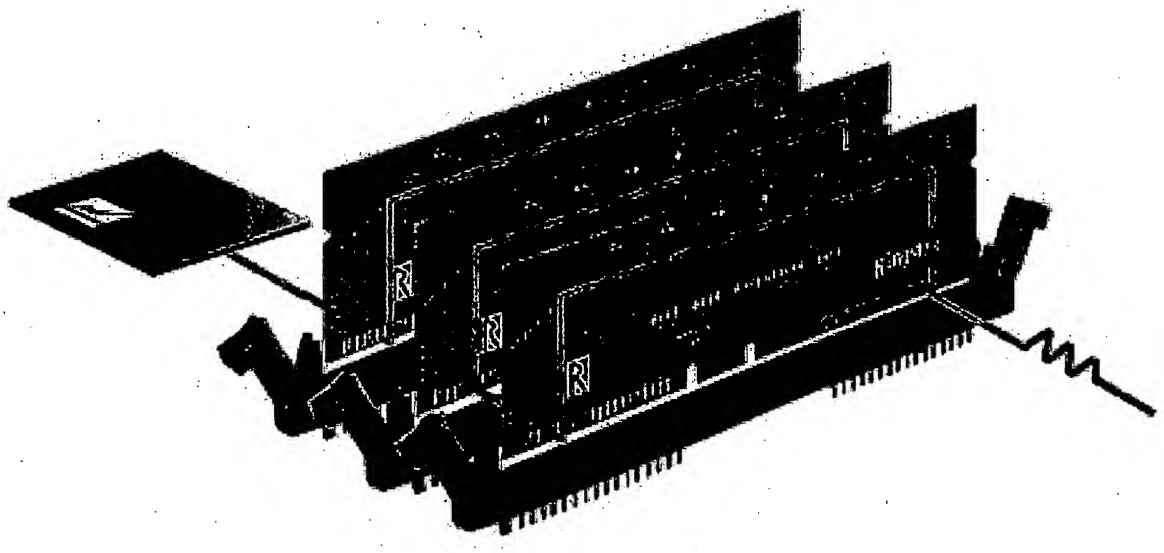

Figure 1: An Example Direct Rambus Memory Subsystem

Introduction

During the last two decades, DRAM technology has progressed dramatically. Device densities have increased from 1 Kbit per chip to 64 Mbits per chip, a factor of 64,000. DRAM performance has not kept pace with these density changes, since access times have decreased by about a factor of 5. Over the same 20 year period, microprocessor performance has jumped by several orders of magnitude. This growing disparity between the speed of microprocessors and that of DRAMs has forced system designers to create a variety of complicated and expensive hierarchical memory techniques, such as SRAM caches and parallel arrays of DRAMs. In addition, now that users demand high performance graphics, systems often rely on expensive frame buffers to provide the necessary bandwidth. And due to the density increases in DRAMs, this need for bandwidth is required from fewer total chips.

To address this processor to memory performance gap, Rambus Inc. has developed a revolutionary chip-to-chip bus, the Direct Rambus™ Channel, that operates up to 10 times faster than conventional DRAMs. The Direct Rambus Channel connects memories to devices such as microprocessors, digital signal processors, graphics processors, and ASICs. The Channel uses a small number of very high speed signals to carry all address, data, and control information. Because it is able to transfer data at 1.6 GBytes per second at a moderate cost, the Direct Rambus Channel is ideal for high performance/low cost systems.

The Rambus solution eliminates the need for buffers and decoders while assuring a modular and scalable system solution. Each Direct Rambus DRAM, called a Direct RDRAM™, transfers data at up to 800 MHz across a two byte-wide Channel. Multiple Channels can be used in parallel to achieve even higher throughput.

The Direct Rambus Channel is implemented using standard PC board layout and manufacturing techniques. Devices connecting to the Direct Channel contain dedicated circuitry, the Rambus Interface, that can be produced on standard sub-micron CMOS processes.

Electrically, the Rambus Channel relies on controlled impedance single terminated transmission lines. These lines carry low-voltage-swing signals. Clock and data always travel in the same direction to virtually eliminate clock to data skew.

Rambus Inc. has assured device independence now and in the future by defining a high-level protocol that allows flexible core configurations. Designing new generations of DRAMs is simplified since the signals comprising the Channel will not change from generation to generation. And, the pipelined nature of the Direct Technology coupled with its ability to handle 16-byte and larger transfers efficiently, Direct Rambus Technology can be applied to a wide variety of applications.

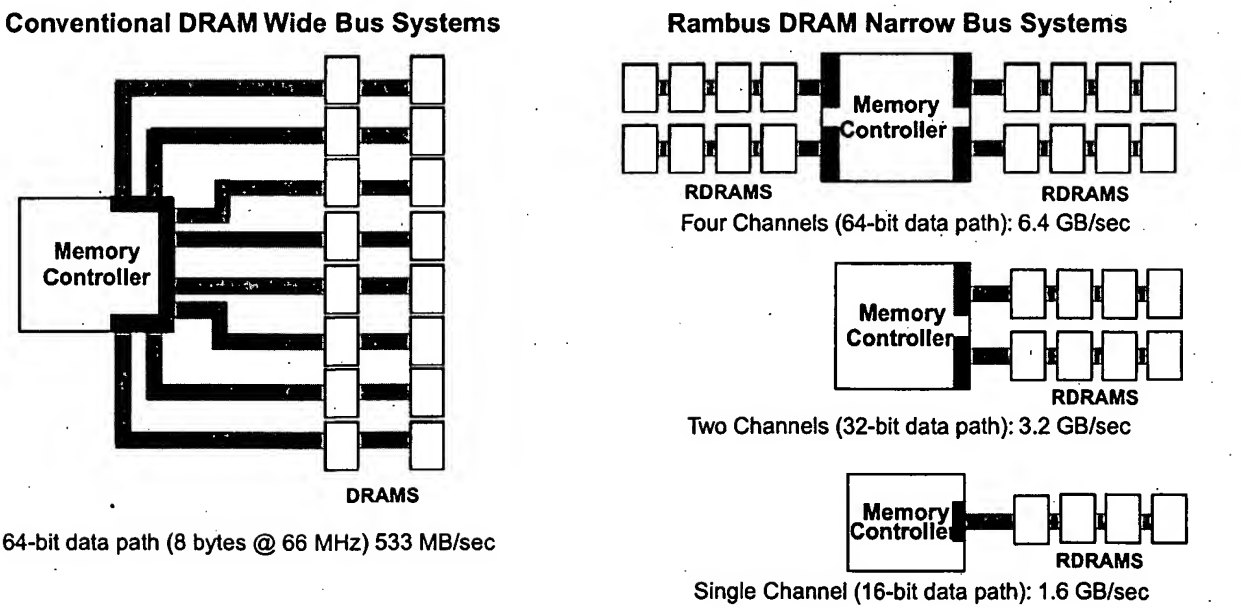

Figure 2: High Bandwidth Memory Solutions

The Advantages of Direct Rambus™ Technology

Direct Rambus memory systems offer a number of significant advantages including high performance, reduced system size and complexity, lower cost and shorter time to market, lower power consumption and improved expandability. These points are summarized in the table below.

High Performance	• 1.6 GBytes per second of bandwidth per Direct Rambus™ Channel • Multiple Channels can be used for even higher performance and bandwidth • An individual Direct RDRAM® offers over ten times higher bandwidth than 66MHz x16 SDRAM device • A 64MByte Direct Rambus system has three times the effective bandwidth over a 64Mbyte 64-bit wide 100MHz SDRAM system
Cost Effective	• Uses conventional DRAM core; The 64Mb Direct RDRAM is comparable in die size to a x16 64M SDRAM • Low cost industry standard memory modules and connectors • Uses industry standard FR-4 printed circuit board technology • Finer granularity allows lower cost memory upgrades
Low Power	• Direct RDRAMs include low power modes that reduce overall power consumption • Rambus devices inherently use less energy per byte transferred • Configurations support 6 times bandwidth of EDO DRAMs at comparable low power
Expandable / Granular	• 32Mb, 64Mb, 128Mb, 256Mb, 512Mb, 1Gb generations of RDRAMs are functionally and electrically compatible. • Memory can be incremented by a single RDRAM • A single Channel supports 32 RDRAMs; Expansion buffers allow support of additional 32 RDRAMs; A controller can support multiple Channels
Reduced Risk; Quick Time to Market	• Fully engineered design, operates at the full rated speed • Proven technology that is shipping in high volume in PCs and consumer products • Support by the leading 13 DRAM suppliers assures OEMs of an ample supply of Direct RDRAMs • Multiple sources for Rambus compatible connectors, modules, clock chips, test systems • A cookbook solution is provided to the system designer
Industry Standard Interface	• All Rambus-based ICs are compatible • Rambus-compatible modules, connectors, clock chips meet Rambus engineering and test specifications • Intel® is developing chip sets for mainstream PCs to ship starting in 1999

Applying the Advantages of Rambus Technology

Direct Rambus technology represents a superior memory solution for a broad range of applications. These applications cover computer, consumer and communication markets. Specific applications include desktop and mobile computers, workstations and servers, graphics/multimedia subsystems and consumer digital video products.

The adoption of Rambus technology offers the potential to dramatically simplify computer architectures, since hardware traditionally used to increase the speed of the processor to memory interface will not be necessary. In addition, the use by Rambus of standard CMOS processes, low cost IC packaging and standard PC board technologies will reduce the expense of building systems based on Rambus technology. These factors will result in faster machines that consume less power, cost less and occupy less physical space.

Desktop Computer Main Memory	• Delivers three times the effective bandwidth of a 64-bit 100MHz SDRAM system • With RIMM modules and connectors fits within the space, cost, power and thermal envelope of a 100MHz SDRAM system • Easily expandable to support multi-gigabyte configurations for high end servers and workstations
Multimedia / Graphics Frame Buffers	• Makes workstation-quality graphics accessible at mass market cost • A single 32Mb RDRAM supports a 1280 x 1024x 24 bits per pixel display with workstation-level 3D graphics performance • Additional RDRAMs can be added easily for greater resolution and more colors • Two Channels per graphics controller for 3.2 gigabytes per second of bandwidth • Provides ample bandwidth to merge the video and graphics frame buffers
Portable Computers	• Built-in power management modes reduce current consumption • Has the lowest power per byte transferred of any DRAM alternative • Can drive internal LCD displays and an external, high resolution monitor • Granularity allows memory expansion in increments of 4 Mbytes
Other Applications	• Workstations • Communications and networking systems • Embedded control • Digital television • Large computers

The Rambus Solution: the Rambus Channel, the RDRAM and the Memory Controller

The Rambus solution replaces costly memory subsystems and interconnect with a single standard high performance chip-to-chip bus and revolutionary DRAMs. This solution has three main elements: the Rambus Interface, the Rambus Channel, and the RDRAM. The Rambus Interface is implemented on both the memory controller and RDRAM devices on the Channel.

The Rambus Channel

The Direct Rambus Channel incorporates a system level specification. Systems using Rambus Channel(s) can operate at full rated speed. This is not the case with systems using conventional DRAMs. Component specifications vary for set-up times, hold times, etc. for DRAMs, ASICs, and buffers. These variations and signal integrity issues combine to slow down cycle times. Also, when DRAM control signals are generated from a master clock, the timing of these signals is often degraded by uneven time of flight, further slowing the system. While the data path is only two bytes wide, the Direct Rambus Channel is capable of transferring data at rates up to 800 MHz.

The Rambus Channel has a well defined mechanical interface. Memory controller and RDRAMs connect to the printed circuit board with an interface that has only 30 high-speed signals. Each Direct Rambus Channel can contain up to 32 RDRAMs. Modular memory expansion is available using RIMM™ modules, in form factors familiar to conventional DRAM implementations.

The Direct Rambus DRAM

The Direct RDRAM is a CMOS DRAM incorporates the Rambus interface circuitry. The Direct Rambus DRAMs are initially available in the 64Mbit generation, although lower density devices such as 32Mbit to very dense 1Gbit density parts will be available over time. RDRAMs are available in both x16 and x18 configurations. The definition and use of the ninth bit per byte is left to the system designer. The Direct RDRAMs use chip scale packaging which is horizontally mounted.

A single RDRAM provides 1.6GBytes per second peak bandwidth. That is over ten times the bandwidth available from a 66MHz x16 SDRAM used in today's PCs. To approach the performance of a single Direct RDRAM, an SDRAM memory system requires a wide, complex, interleaved bus using a large number of SDRAMs.

RDRAMs only respond to requests and therefore require a low level of intelligence. As a result, the memory die size overhead is minimized to keep them highly cost effective.

The Memory Controller

The memory controller contains intelligence and is the only device that generates requests. The Rambus Channel interface can be implemented on conventional microprocessors, peripheral chips, ASIC devices, memory controllers, or graphics chips.

Each memory controller has its own Rambus Interface. The interface cell is available in a wide number of ASIC processes from several vendors. This Interface converts from the low-swing voltage levels used by the Rambus Channel to ordinary CMOS logic levels internal to the ASIC.

All critical, high speed system design issues which arise while designing memory controllers, RDRAMs and the Rambus Channel have been resolved for the designer by Rambus. Designers can implement a Rambus system by using the Company's step-by-step Channel layout documentation.

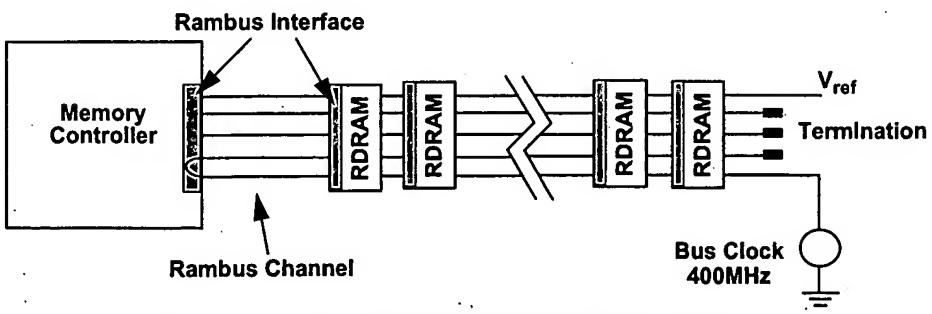

Figure 3: Primary Elements of a Rambus-Based System

The Physical Layer

The high-speed signaling used on the Rambus Channel is called RSL (Rambus Signaling Levels). The Rambus Channel achieves its high-speed operation through a combination of techniques. These include: dense packaging, high quality transmission lines, low-voltage signaling, channel topology, even/odd input samplers, pseudo-differential inputs, differential clocks, current mode drivers, active current control and precise clocking. By employing these techniques on conventional CMOS IC and printed circuit board processes, Rambus technology achieves high performance at low cost.

Dense packaging is essential. RDRAMs use a surface mount chip scale package. This package allows RDRAMs to be spaced closely, densely packing the memory subsystem while keeping the underlying wires short.

This package has a number of electrical advantages. The packages allow a minimal signal stub length. High signal quality is maintained through the reduced stub inductance and low input capacitance low.

Signaling

A Rambus Channel contains 30 high speed, controlled impedance, matched transmission lines:

- ClockToMaster (and its complement ClockToMasterN);
- ClockFromMaster (and its complement ClockFromMasterN);
- Data bus: DQA[8:0], DQB[8:0];
- Address and Control bus: ROW[2:0], COL[4:0].

These high-speed signals are terminated at their characteristic impedance at the RDRAM end of the Channel. As shown in Figure 4, the Rambus Channel has a bus topology with the memory controller at one end, terminators at the other end, and the RDRAMs in between.

The terminators pull the signals up to the system-supplied V_{term} voltage, which corresponds to logic 0. A memory controller or RDRAM asserts a logic 1 by sinking current from the wire, using an open-drain NMOS transistor structure. Each device on the Rambus Channel adjusts its output current drive automatically to keep the signal swing nominal. Current mode signaling provides the additional advantage of having a high-impedance so that signals reflected from the memory controller do not reflect back as they travel back past the driver.

Power is dissipated on the Rambus Channel only when a device drives a logic "1" (low-voltage) on the pin. For typical data patterns, a mixture of ones and zeros on the bus reduces the total power consumption.

All high-speed signals on the Rambus Channel use low voltage swings of 800 mV. Figure 5 shows the nominal voltages of V_{term}, the DC reference, V_{ref}, and the logic 1 level, V_{OL}. Figure 4 shows that V_{ref} may be generated with a resistive divider.

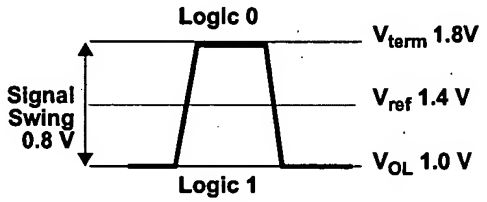

Figure 5: High-Speed Signals Use Low Voltage Swings About Vref

V_{ref} sets the logic threshold for the high-speed low-swing signals. This provides immunity from common mode noise on the Channel. As shown in Figure 4, V_{ref} connects to each device. All devices receive the low-swing signals with dual odd/even differential input circuits and use V_{ref} to set the logic threshold.

This differential sensing allows the Channel to use a low voltage swing. Each of the inputs consists of a pair of differential clock samplers, one operating on the rising edge of clock and the other on the falling edge. The negative input of the input samplers is connected to V_{ref}. Low-voltage-swing signals minimize dv/dt and thus di/dt to provide the following advantages:

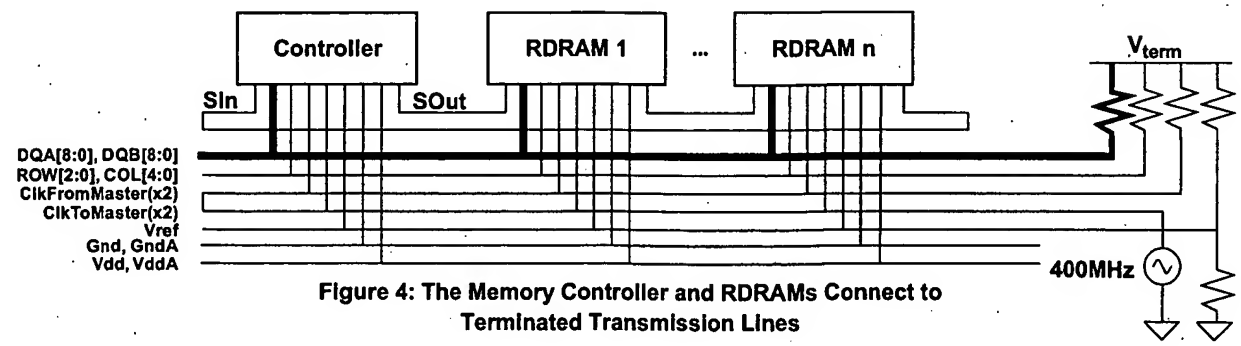

Figure 4: The Memory Controller and RDRAMs Connect to Terminated Transmission Lines

- Reduced ground bounce;
- Reduced power consumption;
- Reduced electromagnetic interference;
- Compatibility with low Vdd devices.

The Channel topology contributes to the high-speed capabilities by ensuring that all signals on the Rambus Channel have the same propagation characteristics and travel across a uniform, parallel layout. This includes the clocks.

Clocking

The Rambus Channel is synchronous, meaning that all commands and data are referenced to clock edges. At Rambus frequencies, special care must be taken to minimize clock to data skew. At the physical level, data is only transferred across the DQA and DQB lines, and all control information is sent across the ROW and COL pins.

Figure 4 shows the clock distribution. The clock source can be a separate clock generator as shown or can be integrated in the memory controller. The clock loop begins at the termination end of the Channel and propagates to the controller end as ClockToMaster, where it loops back as ClockFromMaster to the RDRAMs and terminates.

Clock and data travel in parallel to minimize skew. So, an RDRAM sends data to the memory controller synchronously with ClockToMaster, and the controller sends data to the RDRAMs synchronously with ClockFromMaster. Because the transmission lines are matched, the clock and data signals remain synchronized as they continue to their destination.

Data Transfers

Data transfers occur only between the memory controller and the RDRAMs, and never directly between RDRAMs. Thus signals are terminated at one end of the Channel (Figure 4). Data driven by the controller propagates past all RDRAMs with the desired voltage swing. So, all RDRAMs correctly sense data driven by the controller. The matched terminator eliminates any reflections.

Data driven by an RDRAM moves in both directions at one-half the desired voltage swing. At the controller end, the half-swing pulse reflects off the open end of the wire, doubling in amplitude. Superposition of waveforms and matched termination at the RDRAM mean that RDRAM-to-controller data transfers take place at full speed and full amplitude. Data is effectively transferred on both edges of a 400MHz clock resulting in a 800 Mbits per second per wire transfer rate. Each data transfer uses a 1.25 nanosecond interval, with two of these intervals per clock period.

The two clock edges provide a natural way to label intervals as even and odd (Figure 6). Even intervals occur during clock falling edges and odd intervals during rising edges, with each clock edge at the midpoint of data. Control and data packets begin on even intervals.

Direct Rambus Technology allows the Channel to support multiple clock domains, allowing the length of the Channel to be up to four clock periods, as measured from the ClockToMaster and ClockFromMaster on the last RDRAM on the Channel. The memory expansion modules take advantage of the multiple clock domains.

In order to allow full interleaving across even the first and last RDRAMs on the Channel, the protocol allows each device to be programmed with a read delay. The controller views the effective read delay as the same from all devices.

Also due to the topology of the Channel, write data can be sent down the Channel in the next clock cycle following a Read transaction. As soon as the Read data reaches the controller, it can issue the Write data without any delay.

However, a write followed by a read transaction must wait a short delay, equal to the round-trip length of the Channel. For short Channels, this is only one Rambus clock cycle, 2.5ns. For the longest Channels, the delay could be four clock cycles, or 10ns.

Conventional DRAMs require that one device stop driving the bus before another device starts, effectively causing a one clock cycle delay for bus turnaround. In 100MHz SDRAM systems, this change in Read/Write direction causes a 10ns clock cycle of delay. In contrast, a one-clock delay in the Rambus system is only 2.5ns.

Signals

In each device, the Rambus Interface has 35 active pins.

- 30 High-speed RSL signals (DQA, DQB, ROW, COL and the clocks);
- 4 CMOS signals used for device initialization;
- Vref, a static high-impedance, low current input
- Vdd, Vterm and Ground pins are DC power pins

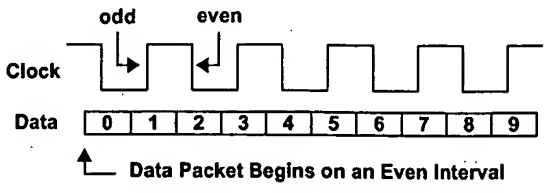

Figure 6: Data Cycle Definitions & Their Relationship to Clock

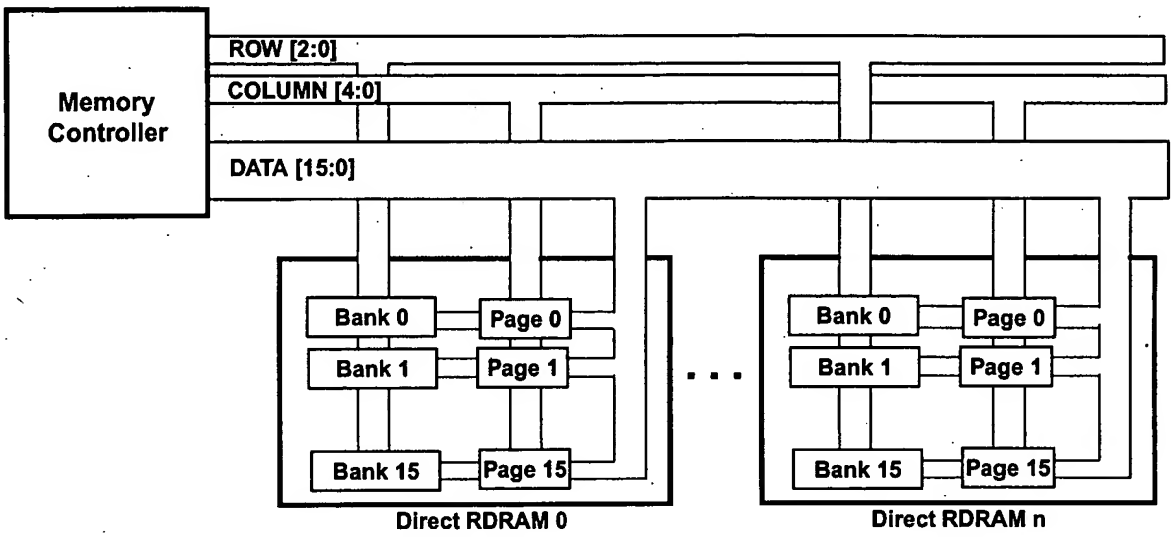

Figure 7: Example System Using the 64M Direct RDRAM

The Logical Layer

On the Rambus Channel, data and control bits move in packets. Each packet type is four clock cycles in length, or 10ns at 800MHz. The Direct Rambus Protocol has been simplified over previous generations of Rambus DRAMs, and the control information is no longer multiplexed on the data bus. A totally independent control and address bus is split into two groups of pins, one for row commands and the other for column commands. Only data is transferred across the two-byte wide data bus.

To ensure proper synchronization of all devices connected to the Rambus Channel, all packets begin during even intervals (falling clock edge). However, packets may begin on any falling clock edge, and may be spaced any number of clock cycles apart.

Each of the buses operate independently of each other, allowing a row command, column command and data to be transferred at the same time to different banks of an RDRAM or to different RDRAMs.

The ROW Packets

Row packets are sent across the three ROW pins of the control and address bus. Row packets include two types of commands: Activate (ACT which is like the falling edge of RAS on an EDO DRAM), or Precharge (PRER or the rising edge of RAS). An Activate command can be sent to any RDRAM bank whose sense-amplifiers have been previously Precharged.

Other commands can be sent across the row pins, including refresh and power state control.

Precharge commands can be sent with an explicit (device, bank) address across the ROW pins. The protocol allows great controller design flexibility by allowing two other forms of Precharge operations on the COL pins.

The Column (COL) Packets

Column packets are sent across the five COL pins of the control and address bus. The COL packets are split into two fields. The first field specifies the primary operation, such as a Read or Write. The second field can be either masks, for Writes, or can be an extended operation (XOP) command. In mask operations, a bit for each byte (a byte may be either 8 or 9 bits) is transferred per 16-byte data packet, totaling 16 mask bits.

Several XOP commands can be used in lieu of the mask field. The most common XOP is an explicit Precharge (PREX). The primary operation of the COL packet can also encode an auto-precharge command along with the Read or Write, allowing an implicit precharge to occur as soon as the requested column operation is completed.

Although the COL packet is composed of the two non-symmetrical fields described above, a COL packet is actually sent split, with the intended command sent first, and the Mask/XOP sent eight clocks (packet size) later. The four-clock delay reduces the amount of intermediate latches necessary to hold the information if it was sent with the COL packet initially.

The Data Packets

Each data packet contains 16 bytes of data. The Direct RDRAMs use a 128-bit wide internal data path, allowing 16 bytes of data to be transferred for each column access.

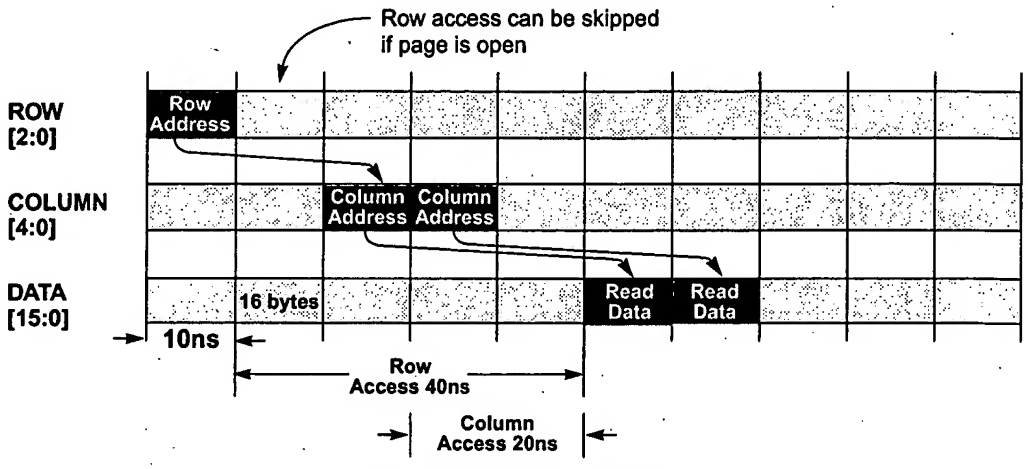

Figure 8: Read Transaction

On the two-byte wide Channel, two bytes are transferred on each rising and falling clock edge, so 16 bytes are transferred in four clock cycles, or 10 ns. In this way, data packets have the same length as row and column packets. Of course, Writes can be masked to allow as little as a single byte to be written at a time.

Since the column bus has as much bandwidth as the data bus, burst commands or burst order information are not needed. This helps to simplify the logic on the RDRAM relative to conventional DRAMs. The controller simply issues multiple COL Read or Write commands in whatever order to any RDRAM, and to any open page.

Read Operation

A Read operation is shown above in Figure 8. Generally, a completely random access is accomplished by the assertion of the ACT command across the ROW pins followed by a Read command sent across the COL pins. After the data is read, a Precharge command (in one of its three forms) is executed to prepare that bank for another completely random Read. Some controller designs may leave activated pages open, anticipating a return to the open page. In that case, the access time is decreased to just the latency of the Read command on the COL pins.

Data is always returned in a fixed, but user selectable number of clock cycles from the end of the Read command. This flexibility in varying the read delay is to accommodate long Channels, where multiple clock domains may be present. In this case, RDRAMs located near to the controller are programmed with the longest latency, and those farthest away are programmed with the shortest latency. This is done during initialization when the position of each RDRAM on the Channel is enumerated. Along with the Device ID (for address selection), the read delay is set according to the RDRAM's clock domain.

Write Operation - Data Transfer Matches Read Data Transfer

Write transaction timing is very much like Read transactions. The control packets are sent in the same way as the Read command. However, one significant difference between a Direct RDRAM and a conventional DRAM is that Write data is delayed to match the timing of a Read transaction's data transfer.

The delay for the Write data was chosen to maximize the usable bandwidth on the data pins. On a conventional SDRAM, when a Read command follows a Write command, there is a gap on the data bus from the Write data to the Read data. This gap is equal to the Read latency of the device. If a 100MHz SDRAM has a Read latency of three cycles, then a 30 ns gap is required after the Write before the Read. The Direct RDRAM avoids this gap by sending the data later in time, and then actually performing the Write subsequent to the data transfer.

A Write command on the COL bus tells the RDRAM that data will be written to the device in an exact number of clock cycles later. Normally, this data would then be written to the core as soon as the data is received.

There is an option to delay this Write and allow a Read command to the same device to "go around" the Write, effectively shortening the read delay, which is usually critically needed. Most Write data on the other hand is usually buffered and is not latency sensitive. If a controller chooses to implement this feature, a check of a single address is all that is needed to make sure that the subsequent Read was not for pending Write data. In that case, the Read can be delayed to its normal time or the data can

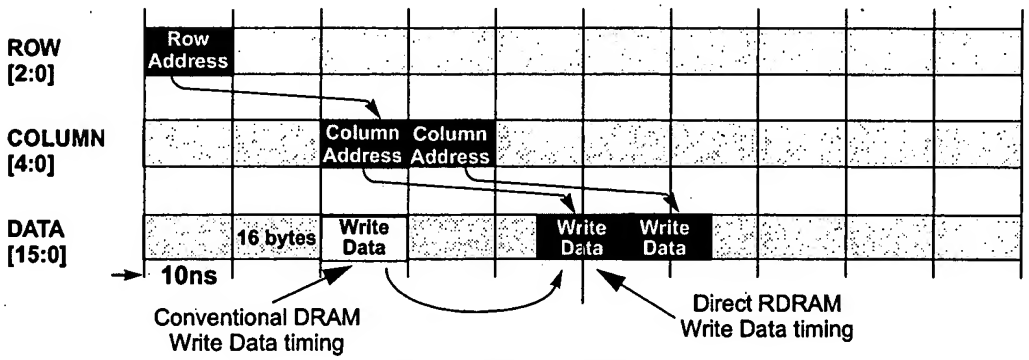

Figure 9: Write Transaction

be returned to the requestor internal to the controller, avoiding the bandwidth of such a trivial transaction on the Channel.

Pipelined Operation

Each of the commands on the control bus may be pipelined, allowing much higher throughput.

The ACT commands can completely absorb the ROW pins, allowing for very fine grain 16-byte random transfers to occur. At the transfer size per Activate increases, the frequency of Activate commands is reduced.

In order to completely fill the data bus, column command would be continuously sent on the COL pins. Except for small gaps of 1 to 3 clocks required for bus turn-around going from a Write to a Read, these busses can be fully utilized.

Two Reads are followed by two Writes to the same device yields over 86% efficiency. An example of reads and writes to multiple RDRAMs can achieve 95% efficiency as shown in the figure below.

Inside a Direct Rambus DRAM (Direct RDRAM)

The following features of the Direct RDRAM allow extremely high peak and effective bandwidth:

- Two-byte wide data bus at 800MHz, yielding a peak 1.6GB/sec data transfer rate;
- Separate control and address bus, further subdivided into ROW and COL buses that work simultaneously and independently;
- Delayed Write mechanism;
- Minimal component and system latency;
- Reduced bank-to-bank interference.

These features allow the Direct Rambus protocol to achieve greater than 95% efficiency for 32-byte random transfers.

DRAM Core Options

Internally, Direct RDRAMs are designed to permit a wide variation in core implementations. The design space allows up to 1Gb DRAM densities, up to 32 RDRAMs per Channel, and enough flexibility in the bank, row and column bits to allow all possible configurations in these densities.

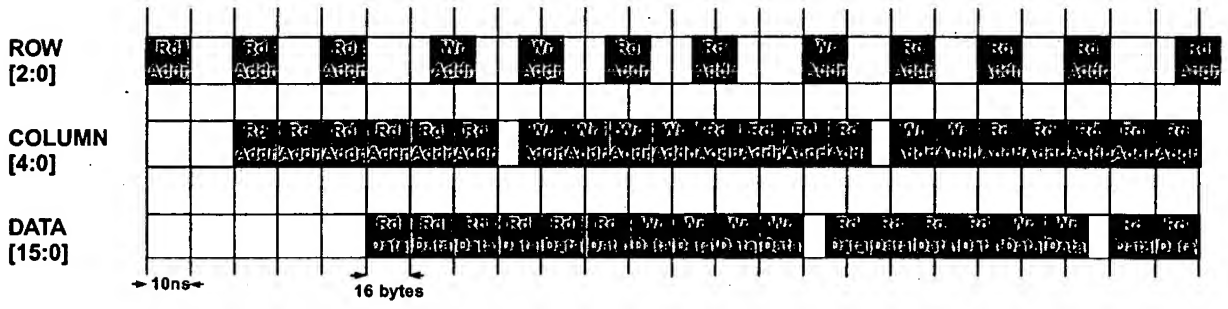

Figure 10: Simultaneous Pipelined Transaction

The Direct RDRAM's internal cores use a 128-/144-bit wide data path operating at 100MHz, which is 1/8th the frequency of the Channel. Thus, every 10 ns, 16-bytes can be transferred to or from the core. The protocol supports a wide variation in core row and column timing parameters.

The 64/72Mbit Direct RDRAMs support either 8 independent or 16 "doubled" banks. The Direct RDRAM core introduces the concept of doubled banks, reducing the necessary sense amplifiers nearly in half while keeping the total number of banks relatively high compared to other DRAM alternatives. The large number of banks helps to prevent interference between memory requests.

The number of banks accessible to the controller is the accumulative number of banks across all the RDRAMs on the Channel. 64Mbyte memory systems implemented with eight 64Mbit RDRAMs or SDRAMs have a wide disparity in the number of banks available to the controller. The RDRAM system has a total of 64 or 128 banks available for access. The 64-bit wide SDRAM system, on the other hand, would have only four banks available if x8 SDRAMs are used.

In an independent bank core, each bank has its own sense amplifiers. In a doubled bank core, sense amplifiers between adjacent banks are shared. The result is that nearly half the sense amplifiers are eliminated, increasing the silicon area efficiency, reducing power consumption and keeping the number of banks high. The restriction imposed by doubled banks is that adjacent banks cannot be activated. Once a bank is activated, that bank must be precharged in order for the adjacent bank to be activated. For greater flexibility, three precharge mechanisms, one on the row pins, and two on the column pins allow the designer great flexibility in determining which RDRAM pages to open or close.

In a doubled bank design, precharging a particular bank will also precharge the adjacent two banks if activated. The hardware state machines in the controller that track the state of the RDRAM array do not to have to check that the adjacent banks are activated. Only the desired bank needs to be precharged before being activated (opened).

Although the cores used in Direct RDRAMs are high-bandwidth they also have low latency, comparable with the fastest SDRAMs used in memory systems. RDRAMs use the same core DRAM technology used to make other types of DRAMs, leveraging significant investments made by the semiconductor manufacturers to minimize the cost of storing data on a piece of silicon.

For low power applications these cores also support self-refresh, permitting the RDRAM to go into a very low power state, and yet retaining all the information in the core during these times.

Direct RDRAM Interface

Direct RDRAMs have separate data and control buses. The data bus permits data transfer rates up to 800MHz, yielding 1.6GB/s on either a x16 or x18 bus configuration. The wider x18 interface is general purpose in nature, allowing system designers the flexibility to use the 9th bit per byte as an error correction mechanism, wider data path or overlay in graphics.

The control bus adds another 800Mb/s of control information to the RDRAM. That is over three times the control information encoded on the control pins of a 100MHz SDRAM or 100MHz DDR SDRAM.

The control bus is further separated into ROW and COL pins, allowing concurrent row and column operations while data is being transferred from a previous command.

The power supply voltage of all Direct RDRAMs has been reduced to 2.5V, further reducing power consumptions and allowing compatibility with future lower voltage devices. The physical layer limits the logic levels of RSL to 1.8V, allowing these RDRAMs to interface directly to ASICs that have very low V_{dd}. The combination of a low V_{dd} and low V_{term} allow the Rambus interface to scale to lower voltage processes, providing greater longevity for the interface.

Power Management

In order to allow lower power system operation, the RDRAM has several operating modes; Active, Standby, Nap, and PowerDown. The four modes are distinguished by two factors, their power consumption, and the time that it takes the RDRAM to execute a transaction from that mode.

In Active mode, the RDRAM is ready to immediately service a transaction. Power consumption is also higher in Active mode than in the other three modes.

Unlike conventional DRAM memory systems, where each device in an entire bank of memory must consume Read/Write power through an entire access, Rambus memory systems use only one device to perform the Read or Write transfer, while all others revert to a lower power state.

An RDRAM automatically transitions to Standby mode at the end of a transaction. Once the device address in the Request packet has been decoded, all of the RDRAMs return to Standby mode with the exception of the one responding to the Request. That device returns to Standby mode once the Read or Write operation is complete.

Alternatively, an RDRAM may be forced to make a standby to active transition. This alternative permits power savings since only the addressed RDRAM moves to the active mode, while the other RDRAMs remain in Standby mode.

Power consumption can be reduced by placing one or more RDRAMs into Nap mode. Nap mode uses less power than standby mode, but takes less time to transition to the Active mode, than PowerDown mode. Systems can achieve a large power saving by keeping the RDRAMs in Nap mode whenever they are not performing a Read or Write transaction.

Power consumption can be further reduced by placing one or more RDRAMs into PowerDown mode.

An example of where these modes can be used is in a portable computer application. Here, sleep mode is implemented by placing a majority of the RDRAMs in PowerDown, while the RDRAM that contains the frame buffer is placed in Nap mode. This permits screen refresh to occur without powering up the entire memory system.

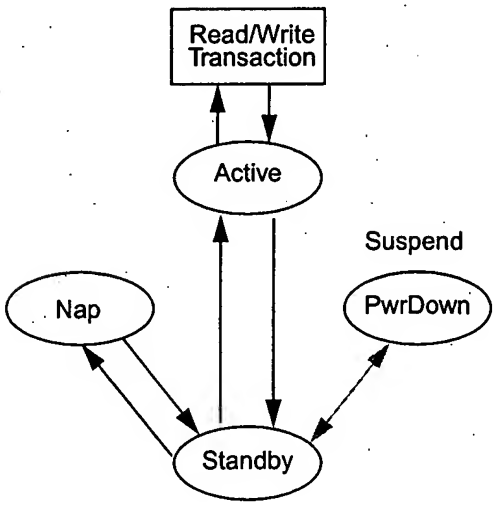

Figure 8: Power Transitions in the Direct RDRAM

Rambus Memory Controller

The Rambus Memory Controller ("RMC") is a block of digital logic residing on a Rambus-based controller to drive and manage the memory transactions of a Rambus memory system. The interface to the RMC is a simple two-wire handshake. The RMC directly connects to the Rambus ASIC Cell (RAC) as an Input/Output Cell. The RAC provides the basic multiplexing/ demultiplexing functions for converting from a byte-serial bus operating at the Channel frequency—up to 800MHz—to the controller's eight-byte wide bus with an up to 200 MHz signaling rate. The RAC manages the electrical and physical interface to the Rambus subsystem.

The RMC provides the protocol for performing Read and Write transactions to the Rambus DRAMs. The RMC is provided as synthesizeable Verilog or VHDL source code which can be incorporated in the controller design process. The RMC code and companion Guide serve as a reference design for system customers.

Features of the Rambus Memory Controller

- Complete Direct Rambus protocol support. The Rambus Memory Controller translates address, data and command into the Rambus protocol. It presents a simplified, high level 128/144-bit data path to the controller designer.
- Complete memory control support. The RMC supports all control functions including protocol, refresh, memory and interleaving support.
- Simple two-wire handshake interface.
- Fully simulated and verified.
- Compatibility. The RMC works with all Rambus DRAM and RAC-based controller devices.
- Implemented in Verilog or VHDL. Easy to use in any controller design process to be used with ASIC design or COT design programs.
- The RMC synthesizes to no more than 14K gates. Can be configured for multiple Rambus Channels.

The Rambus Memory Controller consists of synthesizeable Verilog or VHDL source code to support Rambus memory transactions. It is a gate-level description of the RMC. Only low level library functions, such as can be built from AND and OR functions, are needed to support the RMC. It accepts Read and Write transactions from the controller and manages the operation of the RAC and RDRAMs in an optimal manner; no Channel bandwidth is wasted and the RMC adds minimal latency to Read transactions. The RMC also supports interleaved transactions,

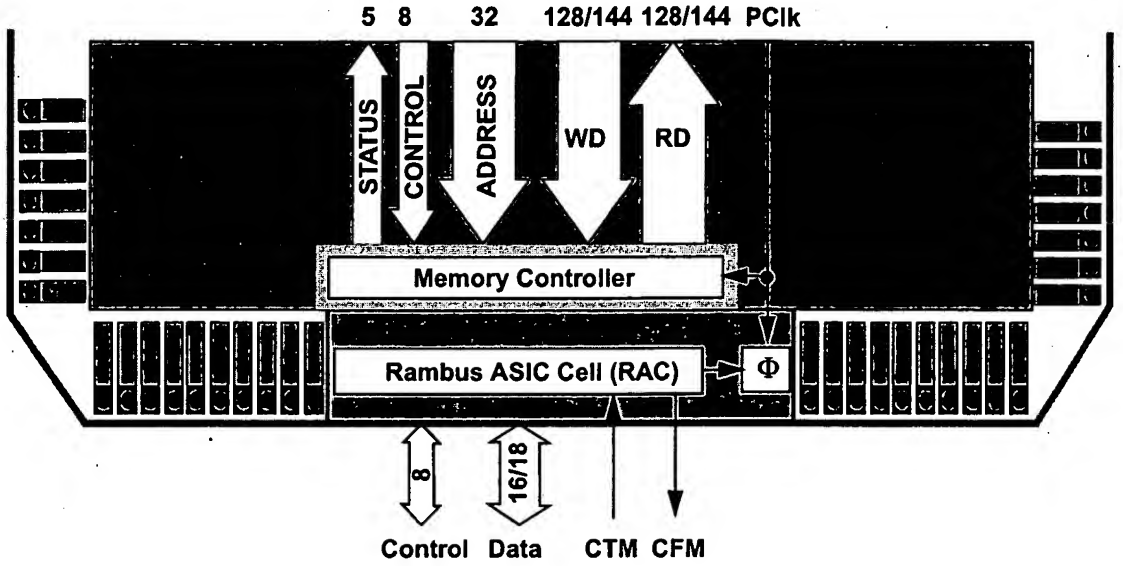

Figure 9: Direct Rambus Memory Controller

permitting a RAS access to be started in one RDRAM while a CAS access is performed to another. The RMC code may be used as-is or be modified for a particular system application.

System Packaging

The Direct Rambus memory system uses industry standard packaging that can be supported on standard construction printed circuit boards. Systems incorporating Rambus components are compact, with well controlled electrical signaling and a simple, straight line Printed Circuit Board (PCB) topology. The PCB layout is straightforward.

The Direct RDRAM components use chip scale packaging, which offers excellent electrical characteristics, including low capacitance and thermal design. The chip scale package is about the same size as the RDRAM die, uses a minimum of material and thus has potential for the lowest cost chip packaging.

The memory upgrade packaging consists of the Rambus RIMM™ memory module and connector which are similar in size to existing DIMM memory modules and connectors. The RIMM modules support up to eight Direct RDRAMs on each side of the module. A single RIMM module can accommodate up to 128 Mbytes of memory using 64-Mbit RDRAM devices.

Tailored to the system memory market, the RIMM is comparable in fit, form factor, and cost to SDRAM DIMMs, and will offer three times the bandwidth with lower power consumption and thermal properties. A 64-Mbyte RIMM module will deliver 1.5-Gbyte/sec performance under an average workload consisting of 30% Writes and random 32-byte transfers.

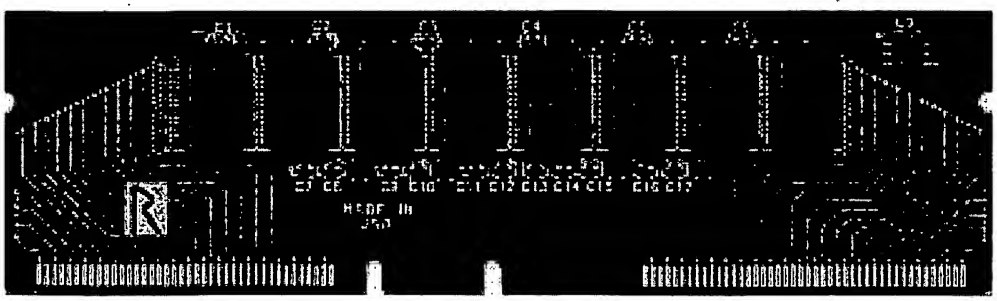

Figure 10: Physical Organization of a Rambus-Based System

The RIMM is compatible with standard motherboard form factors including Intel's ATX, NLX, and mini-NLX chassis. The motherboard can support up to three module sockets. The Direct Rambus Channel signals are daisy chained through each module.

The module design allows the first RIMM connector to be placed up to six inches from the memory controller.

The Direct Rambus memory upgrade module can support 32Mb, 64Mb, 128Mb, 256Mb, 512Mb or 1 Gb density RDRAMs.